

# Orthonormal Expansion $\ell_1$ -Minimization Algorithms for Compressed Sensing

Zai Yang, Cishen Zhang, Jun Deng, and Wenmiao Lu

**Abstract**—Compressed sensing aims at reconstructing sparse signals from significantly reduced number of samples, and a popular reconstruction approach is  $\ell_1$ -norm minimization. In this correspondence, a method called orthonormal expansion is presented to reformulate the basis pursuit problem for noiseless compressed sensing. Two algorithms are proposed based on convex optimization: one exactly solves the problem and the other is a relaxed version of the first one. The latter can be considered as a modified iterative soft thresholding algorithm and is easy to implement. Numerical simulation shows that, in dealing with noise-free measurements of sparse signals, the relaxed version is accurate, fast and competitive to the recent state-of-the-art algorithms. Its practical application is demonstrated in a more general case where signals of interest are approximately sparse and measurements are contaminated with noise.

**Index Terms**—Augmented Lagrange multiplier, Compressed sensing,  $\ell_1$  minimization, Orthonormal expansion, Phase transition, Sparsity-undersampling tradeoff.

## I. INTRODUCTION

COMPRESSED sensing (CS) aims at reconstructing a signal from its significantly reduced measurements with a priori knowledge that the signal is (approximately) sparse [1]–[3]. In CS, the signal  $\mathbf{x}^o \in \mathbb{R}^N$  of interest is acquired by taking measurements of the form

$$\mathbf{b} = \mathbf{A}\mathbf{x}^o + \mathbf{e},$$

where  $\mathbf{A} \in \mathbb{R}^{n \times N}$  is the sampling matrix,  $\mathbf{b} \in \mathbb{R}^n$  is the vector of our measurements or samples,  $\mathbf{e} \in \mathbb{R}^n$  is the measurement noise, with  $N$  and  $n$  being sizes of the signal and acquired samples, respectively. A standard approach to reconstructing the original signal  $\mathbf{x}^o$  is to solve

$$(\text{BP}_\epsilon) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1, \text{ subject to } \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2 \leq \epsilon,$$

which is known as the basis pursuit denoising (BPDN) problem, with  $\|\mathbf{e}\|_2 \leq \epsilon$ . Another frequently discussed approach is to solve the problem in its Lagrangian form

$$(\text{QP}_\lambda) \quad \min_{\mathbf{x}} \left\{ \lambda \|\mathbf{x}\|_1 + \frac{1}{2} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 \right\}.$$

To appear in IEEE Transactions on Signal Processing. Copyright (c) 2011 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

Z. Yang, J. Deng and W. Lu are with the School of Electrical and Electronic Engineering, Nanyang Technological University, 639798, Singapore (e-mail: yang0248@e.ntu.edu.sg; de0001un@e.ntu.edu.sg; wenmiao@ntu.edu.sg).

C. Zhang is with the Faculty of Engineering and Industrial Sciences, Swinburne University of Technology, Hawthorn VIC 3122, Australia (e-mail: cishenzhang@swin.edu.au).

It follows from the knowledge of convex optimization that  $(\text{BP}_\epsilon)$  and  $(\text{QP}_\lambda)$  are equivalent with appropriate choices of  $\epsilon$  and  $\lambda$ . In general,  $\lambda$  decreases as  $\epsilon$  decreases. In the limiting case of  $\lambda, \epsilon \rightarrow 0$ , both  $(\text{BP}_\epsilon)$  and  $(\text{QP}_\lambda)$  converges to the following basis pursuit (BP) problem in noiseless CS:

$$(\text{BP}) \quad \min_{\mathbf{x}} \|\mathbf{x}\|_1, \text{ subject to } \mathbf{A}\mathbf{x} = \mathbf{b}.$$

It is important to develop accurate and computationally efficient algorithms to deal with the  $\ell_1$  minimization problems of high dimensional signals in CS, such as an image of  $512 \times 512$  pixels. One popular approach for solving  $(\text{QP}_\lambda)$  is the iterative soft thresholding (IST) algorithm of the form (stating from  $\mathbf{x}_0 = 0$ ) [4], [5]

$$\mathbf{x}_{t+1} = S_\lambda(\mathbf{x}_t + \mathbf{A}'\mathbf{z}_t), \quad \mathbf{z}_t = \mathbf{b} - \mathbf{A}\mathbf{x}_t, \quad (1)$$

where  $'$  denotes the transpose operator and  $S_\lambda(\cdot)$  is the soft thresholding operator with a threshold  $\lambda$ , which will be defined more precisely in Section II-A. IST has a concise form and is easy to implement, but its convergence can be very slow in some cases [5], especially for small  $\lambda$ . Other algorithms proposed to solve  $(\text{QP}_\lambda)$  include interior-point method [6], conjugate gradient method [7] and fixed-point continuation [8]. Few algorithms can accurately solve large-scale BPDN problem  $(\text{BP}_\epsilon)$  with a low computational complexity. The  $\ell_1$ -magic package [9] includes a primal log barrier code solving  $(\text{BP}_\epsilon)$ , in which the conjugate gradient method may not find a precise Newton step in the large-scale mode. NESTA [10] approximately solves  $(\text{BP}_\epsilon)$  based on Nesterov's smoothing technique [11], with continuation.

In the case of strictly sparse signals and noise-free measurements, many fast algorithms have been proposed to exactly reconstruct  $\mathbf{x}^o$ . One class of algorithms uses the greedy pursuit method, which iteratively refines the support and entries of a sparse solution to yield a better approximation of  $\mathbf{x}^o$ , such as OMP [12], StOMP [13] and CoSaMP [14]. These algorithms, however, may not produce satisfactory sparsity-undersampling tradeoff compared with (BP) because of their greedy operations. As mentioned before,  $(\text{QP}_\lambda)$  is equivalent to (BP) as  $\lambda \rightarrow 0$ . Hence, (BP) can be solved with high accuracy using algorithms for  $(\text{QP}_\lambda)$  by setting  $\lambda$  to a small value, e.g.  $1 \times 10^{-6}$ . IST has attracted much attention because of its simple form. In the case where  $\lambda$  is small, however, the standard IST in (1) can be very slow. To improve its speed, a fixed-point continuation (FPC) strategy is exploited [8], in which  $\lambda$  is decreased in a continuation scheme and a  $q$ -linear convergence rate is achieved. Further, FPC-AS [15] is developed to improve the performance of FPC by introducing

an active set, inspired by greedy pursuit algorithms. An alternative approach to improving the speed of IST is to use an aggressive continuation, which takes the form

$$\mathbf{x}_{t+1} = S_{\lambda_t}(\mathbf{x}_t + \mathbf{A}'\mathbf{z}_t), \quad \mathbf{z}_t = \mathbf{b} - \mathbf{A}\mathbf{x}_t, \quad (2)$$

where  $\lambda_t$  may decrease in each iteration. The algorithm of this form typically has a worse sparsity-undersampling tradeoff than (BP) [16]. Such a disadvantage is partially overcome by approximately message passing (AMP) algorithm [17], in which a modification is introduced in the current residual  $\mathbf{z}_t$ :

$$\mathbf{x}_{t+1} = S_{\lambda_t}(\mathbf{x}_t + \mathbf{A}'\mathbf{z}_t), \quad \mathbf{z}_t = \mathbf{b} - \mathbf{A}\mathbf{x}_t + \frac{N \|\mathbf{x}_t\|_0}{n} \mathbf{z}_{t-1}, \quad (3)$$

where  $\|\mathbf{x}\|_0$  counts the number of nonzero entries of  $\mathbf{x}$ . It is noted that AMP having the same sparsity-undersampling tradeoff as (BP) is only established based on heuristic arguments and numerical simulations. Moreover, it cannot be easily extended to deal with more general complex-valued sparse signals, though real-valued signals are only considered in this correspondence.

This correspondence focuses on solving the basis pursuit problem (BP) in noiseless CS. We assume that  $\mathbf{A}\mathbf{A}'$  is an identity matrix, i.e., the rows of  $\mathbf{A}$  are orthonormal. This is reasonable since most fast transforms in CS are of this form, such as discrete cosine transform (DCT), discrete Fourier transform (DFT) and some wavelet transforms, e.g. Haar wavelet transform. Such an assumption has also been used in other algorithms, e.g. NESTA. A novel method called orthonormal expansion is introduced to reformulate (BP) based on this assumption. The exact OrthoNormal Expansion  $\ell_1$  minimization (eONE-L1) algorithm is then proposed to exactly solve (BP) based on the augmented Lagrange multiplier (ALM) method, which is a convex optimization method.

The relaxed ONE-L1 (rONE-L1) algorithm is further developed to simplify eONE-L1. It is shown that rONE-L1 converges at least exponentially and is in the form of modified IST in (2). In the case of strictly sparse signals and noise-free measurements, numerical simulations show that rONE-L1 has the same sparsity-undersampling tradeoff as (BP) does. Under the same conditions, rONE-L1 is compared with state-of-the-art algorithms, including FPC-AS, AMP and NESTA. It is shown that rONE-L1 is faster than AMP and NESTA when the number of measurements is just enough to exactly reconstruct the original sparse signal using  $\ell_1$  minimization. In a general case of approximately sparse signals and noise-contaminated measurements, where AMP is omitted for its poor performance, an example of 2D image reconstruction from its partial-DCT measurements demonstrates that rONE-L1 outperforms NESTA and FPC-AS in terms of computational efficiency and reconstruction quality, respectively.

The rest of the correspondence is organized as follows. Section II introduces the proposed exact and relaxed ONE-L1 algorithms followed by their implementation details. Section III reports the efficiency of our algorithm through numerical simulations in both noise-free and noise-contaminated cases. Conclusions are drawn in Section IV.

## II. ONE-L1 ALGORITHMS

### A. Preliminary: Soft Thresholding Operator

For  $w \in \mathbb{R}$ , the soft thresholding of  $w$  with a threshold  $\lambda \in \mathbb{R}^+$  is defined as:

$$S_\lambda(w) = \text{sgn}(w) \cdot (|w| - \lambda)^+,$$

where  $(\cdot)^+ = \max(\cdot, 0)$  and

$$\text{sgn}(w) = \begin{cases} w/|w|, & w \neq 0; \\ 0, & w = 0. \end{cases}$$

The operator  $S_\lambda(\cdot)$  can be extended to vector variables by its element-wise operation.

The soft thresholding operator can be applied to solve the following  $\ell_1$ -norm regularized least square problem [4], i.e.,

$$S_\lambda(\mathbf{w}) = \arg \min_{\mathbf{v}} \left\{ \lambda \|\mathbf{v}\|_1 + \frac{1}{2} \|\mathbf{w} - \mathbf{v}\|_2^2 \right\}. \quad (4)$$

where  $\mathbf{v}, \mathbf{w} \in \mathbb{R}^n$ , and  $S_\lambda(\mathbf{w})$  is the unique minimizer.

### B. Problem Description

Consider the  $\ell_1$ -minimization problem (BP) with the sampling matrix  $\mathbf{A}$  satisfying that  $\mathbf{A}\mathbf{A}' = \mathbf{I}$ , where  $\mathbf{I}$  is an identity matrix. We call that  $\mathbf{A}$  is a partially orthonormal matrix hereafter as its rows are usually randomly selected from an orthonormal matrix in practice, e.g. partial-DCT matrix. Hence, there exists another partially orthonormal matrix  $\mathbf{B} \in \mathbb{R}^{(N-n) \times N}$ , whose rows are orthogonal to those of  $\mathbf{A}$ , such that  $\Phi = \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}$  is orthonormal. Let  $\mathbf{p} = \Phi\mathbf{x}$ . The problem (BP) is then equivalent to

$$(\text{BP}^o) \quad \min_{\mathbf{x}, \mathbf{p}, \Gamma(\mathbf{p})=\mathbf{b}} \|\mathbf{x}\|_1, \quad \text{subject to } \Phi\mathbf{x} = \mathbf{p},$$

where  $\Gamma(\mathbf{p})$  is an operator projecting the vector  $\mathbf{p}$  onto its first  $n$  entries.

In  $(\text{BP}^o)$ , the sampling matrix  $\mathbf{A}$  is expanded into an orthonormal matrix  $\Phi$ . It corresponds to the scenario where the full sampling is carried out with the sampling matrix  $\Phi$  and  $\mathbf{p}$  is the vector containing all measurements. Note that only  $\mathbf{b}$ , as a part of  $\mathbf{p}$ , is actually observed. To expand the sampling matrix  $\mathbf{A}$  into an orthonormal matrix  $\Phi$  is a key step to show that the ALM method exactly solves  $(\text{BP}^o)$  and, hence, (BP). The next subsection describes the proposed algorithm, referred to as orthonormal expansion  $\ell_1$ -minimization.

### C. ALM Based ONE-L1 Algorithms

In this subsection we solve the  $\ell_1$ -minimization problem  $(\text{BP}^o)$  using the ALM method [18], [19]. The ALM method is similar to the quadratic penalty method except an additional Lagrange multiplier term. Compared with the quadratic penalty method, ALM method has some salient properties, e.g. the ease of parameter tuning and the convergence speed. The augmented Lagrangian function is

$$\mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu) = \|\mathbf{x}\|_1 + \langle \mathbf{p} - \Phi\mathbf{x}, \mathbf{y} \rangle + \frac{\mu}{2} \|\mathbf{p} - \Phi\mathbf{x}\|_2^2, \quad (5)$$

where Lagrange multiplier  $\mathbf{y} \in \mathbb{R}^N$ ,  $\mu \in \mathbb{R}^+$  and  $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^\top \mathbf{v} \in \mathbb{R}$  is the inner product of  $\mathbf{u}, \mathbf{v} \in \mathbb{R}^N$ . Eq. (5) can be expressed as follows:

$$\mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu) = \|\mathbf{x}\|_1 + \frac{\mu}{2} \|\mathbf{p} - \Phi \mathbf{x} + \mu^{-1} \mathbf{y}\|_2^2 - \frac{1}{2\mu} \|\mathbf{y}\|_2^2. \quad (6)$$

Subsequently, we have the following optimization problem (SP):

$$(SP) \quad \min_{\mathbf{x}, \mathbf{p}, \Gamma(\mathbf{p})=\mathbf{b}} \mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu).$$

Instead of solving (SP), let us consider the two related problems

$$(SP_1) \quad \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu),$$

and

$$(SP_2) \quad \min_{\mathbf{p}, \Gamma(\mathbf{p})=\mathbf{b}} \mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu).$$

Note that problem (SP<sub>1</sub>) is similar to the  $\ell_1$ -regularized problem in (4). In general, (SP<sub>1</sub>) cannot be directly solved using the soft thresholding operator as that in (4) since there is a matrix product of  $\Phi$  and  $\mathbf{x}$  in the term of  $\ell_2$ -norm. However, the soft thresholding operator does apply to the special case where  $\Phi$  is orthonormal. Given  $\|\Phi \mathbf{u}\|_2 = \|\mathbf{u}\|_2$  for any  $\mathbf{u} \in \mathbb{R}^N$ , we can apply the soft thresholding to obtain

$$S_{\mu^{-1}}(\Phi'(\mathbf{p} + \mu^{-1} \mathbf{y})) = \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu). \quad (7)$$

To solve (SP<sub>2</sub>), we let  $\partial_{\Gamma(\mathbf{p})} \mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu) = 0$  to obtain  $\bar{\Gamma}(\mathbf{p}) = \bar{\Gamma}(\Phi \mathbf{x} - \mu^{-1} \mathbf{y})$ , i.e.,

$$\left[ \bar{\Gamma}(\Phi \mathbf{x} - \mu^{-1} \mathbf{y}) \right] = \arg \min_{\mathbf{p}, \Gamma(\mathbf{p})=\mathbf{b}} \mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu), \quad (8)$$

where  $\bar{\Gamma}(\cdot)$  is the operator projecting the variable to its last  $N - n$  entries. As a result, an iterative solution of (SP) is stated in the following Lemma 1.

*Lemma 1:* For fixed  $\mathbf{y}$  and  $\mu$ , the iterative algorithm given by

$$\mathbf{x}^{j+1} = S_{\mu^{-1}}(\Phi'(\mathbf{p}^j + \mu^{-1} \mathbf{y})), \quad (9)$$

$$\mathbf{p}^{j+1} = \left[ \bar{\Gamma}(\Phi \mathbf{x}^{j+1} - \mu^{-1} \mathbf{y}) \right] \quad (10)$$

converges to an optimal solution of (SP).

*Proof:* Denote  $\mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}, \mu)$  as  $\mathcal{L}(\mathbf{x}, \mathbf{p})$ , for simplicity. By the optimality and uniqueness of  $\mathbf{x}^{j+1}$  and  $\mathbf{p}^{j+1}$ , we have  $\mathcal{L}(\mathbf{x}^{j+1}, \mathbf{p}^{j+1}) \leq \mathcal{L}(\mathbf{x}^j, \mathbf{p}^j)$  and the equality holds if and only if  $(\mathbf{x}^{j+1}, \mathbf{p}^{j+1}) = (\mathbf{x}^j, \mathbf{p}^j)$ . Hence, the sequence  $\{\mathcal{L}(\mathbf{x}^j, \mathbf{p}^j)\}$  is bounded and converges to a constant  $L^*$ , i.e.,  $\mathcal{L}(\mathbf{x}^j, \mathbf{p}^j) \rightarrow L^*$  as  $j \rightarrow +\infty$ . Since the sequence  $\{\mathbf{x}^j\}$  is also bounded by  $\|\mathbf{x}^j\|_1 \leq \mathcal{L}(\mathbf{x}^j, \mathbf{p}^j) + \frac{1}{2\mu} \|\mathbf{y}\|_2^2$ , there exists a sub-sequence  $\{\mathbf{x}^{j_i}\}_{i=1}^{+\infty}$  such that  $\mathbf{x}^{j_i} \rightarrow \mathbf{x}_s^*$  as  $i \rightarrow +\infty$ , where  $\mathbf{x}_s^*$  is an accumulation point of  $\{\mathbf{x}^j\}$ . Correspondingly,  $\mathbf{p}^{j_i} \rightarrow \mathbf{p}_s^*$ , and moreover,  $\mathcal{L}(\mathbf{x}_s^*, \mathbf{p}_s^*) = L^*$ .

We now use contradiction to show that  $(\mathbf{x}_s^*, \mathbf{p}_s^*)$  is a fixed point of the algorithm. Suppose that there exist  $(\bar{\mathbf{x}}_s, \bar{\mathbf{p}}_s) \neq (\mathbf{x}_s^*, \mathbf{p}_s^*)$  such that  $\bar{\mathbf{x}}_s = \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{p}_s^*)$  and  $\bar{\mathbf{p}}_s = \arg \min_{\mathbf{p}, \Gamma(\mathbf{p})=\mathbf{b}} \mathcal{L}(\mathbf{x}_s^*, \mathbf{p})$ , and  $\mathcal{L}(\bar{\mathbf{x}}_s, \bar{\mathbf{p}}_s) < L^*$ . By (9), (10) and [4, Lemma 2.2], we have  $\|\mathbf{x}^{j_i+1} - \bar{\mathbf{x}}_s\|_2 \leq \|\mathbf{x}^{j_i} - \mathbf{x}_s^*\|_2 \rightarrow 0$ , i.e.,  $\mathbf{x}^{j_i+1} \rightarrow \bar{\mathbf{x}}_s$ , as  $i \rightarrow +\infty$ . Meanwhile,

$\mathbf{p}^{j_i+1} \rightarrow \bar{\mathbf{p}}_s$ . Hence,  $\mathcal{L}(\mathbf{x}^{j_i+1}, \mathbf{p}^{j_i+1}) \rightarrow \mathcal{L}(\bar{\mathbf{x}}_s, \bar{\mathbf{p}}_s) < L^*$ , which contradicts  $\mathcal{L}(\mathbf{x}^j, \mathbf{p}^j) \rightarrow L^*$ , resulting in that  $(\mathbf{x}_s^*, \mathbf{p}_s^*)$  is a fixed point. Moreover, it follows from  $\|\mathbf{x}^{j_i+q} - \mathbf{x}_s^*\|_2 \leq \|\mathbf{x}^{j_i} - \mathbf{x}_s^*\|_2 \rightarrow 0$  for any positive integer  $q$ , that  $\mathbf{x}^j \rightarrow \mathbf{x}_s^*$ , as  $j \rightarrow +\infty$ .

Note that orthonormal matrix  $\Phi = \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix}$  and  $\Phi' \Phi = \mathbf{A}' \mathbf{A} + \mathbf{B}' \mathbf{B} = \mathbf{I}$ . We can obtain

$$\mathbf{x}_s^* = S_{\mu^{-1}}(\mathbf{x}_s^* + \mathbf{A}'(\mathbf{b} + \mu^{-1} \Gamma(\mathbf{y}) - \mathbf{A} \mathbf{x}_s^*)). \quad (11)$$

Meanwhile, (SP) is equivalent to

$$\begin{aligned} \min_{\mathbf{x}} \mathcal{L} \left( \mathbf{x}, \left[ \bar{\Gamma}(\Phi \mathbf{x} - \mu^{-1} \mathbf{y}) \right] \right) \\ = \|\mathbf{x}\|_1 + \frac{\mu}{2} \|\mathbf{A} \mathbf{x} - \mathbf{b} - \mu^{-1} \Gamma(\mathbf{y})\|_2^2 - \frac{1}{2\mu} \|\mathbf{y}\|_2^2. \end{aligned} \quad (12)$$

By [4, Proposition 3.10],  $\mathbf{x}_s^*$  is an optimal solution of the problem in (12) and equivalently,  $(\mathbf{x}_s^*, \mathbf{p}_s^*)$  is an optimal solution of (SP). ■

*Remark 1:*

- (1) Lemma 1 shows that to solve problem (SP) is equivalent to solve, iteratively, problems (SP<sub>1</sub>) and (SP<sub>2</sub>).
- (2) Reference [4, Proposition 3.10] only deals with the special case  $\|\mathbf{A}\|_2 < 1$  and it is, in fact, straightforward to extend the result to arbitrary  $\mathbf{A}$ .

Following from the framework of the ALM method [19] and Lemma 1, the ALM based ONE-L1 algorithm is outlined in Algorithm 1, where  $(\mathbf{x}_t^*, \mathbf{p}_t^*)$  is the optimal solution to (SP) in the  $t$ th iteration and  $\mathbf{y}_t^*$  is the corresponding Lagrange multiplier.

Algorithm 1: Exact ONE-L1 Algorithm via ALM Method

Input: Expanded orthonormal matrix  $\Phi$  and observed sample data  $\mathbf{b}$ .

1.  $\mathbf{x}_0^* = \mathbf{0}; \mathbf{p}_0^* = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix}; \mathbf{y}_0^* = \mathbf{0}; \mu_0 > 0; t = 0$ .
  2. while not converged do
  3. Lines 4-9 solve  $(\mathbf{x}_{t+1}^*, \mathbf{p}_{t+1}^*) = \arg \min_{(\mathbf{x}, \mathbf{p}, \Gamma(\mathbf{p})=\mathbf{b})} \mathcal{L}(\mathbf{x}, \mathbf{p}, \mathbf{y}_t^*, \mu_t)$ ;
  4.  $\mathbf{x}_{t+1}^0 = \mathbf{x}_t^*, \mathbf{p}_{t+1}^0 = \mathbf{p}_t^*, j = 0$ ;
  5. while not converged do
  6.  $\mathbf{x}_{t+1}^{j+1} = S_{\mu_t^{-1}}(\Phi'(\mathbf{p}_{t+1}^j + \mu_t^{-1} \mathbf{y}_t^*))$ ;
  7.  $\mathbf{p}_{t+1}^{j+1} = \left[ \bar{\Gamma}(\Phi \mathbf{x}_{t+1}^{j+1} - \mu_t^{-1} \mathbf{y}_t^*) \right]$ ;
  8. set  $j = j + 1$ ;
  9. end while
  10.  $\mathbf{y}_{t+1}^* = \mathbf{y}_t^* + \mu_t(\mathbf{p}_{t+1}^* - \Phi \mathbf{x}_{t+1}^*)$ ;
  11. choose  $\mu_{t+1} > \mu_t$ ;
  12. set  $t = t + 1$ ;
  13. end while
- Output:  $(\mathbf{x}_t^*, \mathbf{p}_t^*)$ .

The convergence of Algorithm 1 is stated in the following theorem.

*Theorem 1:* Any accumulation point  $(\mathbf{x}^*, \mathbf{p}^*)$  of sequence  $\{(\mathbf{x}_t^*, \mathbf{p}_t^*)\}_{t=1}^{+\infty}$  of Algorithm 1 is an optimal solution of (BP<sup>o</sup>) and the convergence rate with respect to the outer iteration loop index  $t$  is at least  $O(\mu_{t-1}^{-1})$  in the sense that

$$\|\mathbf{x}_t^* - \mathbf{x}^\dagger\|_1 = O(\mu_{t-1}^{-1}),$$

where  $\mathbf{x}^\dagger = \|\mathbf{x}^*\|_1$ .

*Proof:* We first show that the sequence  $\{\mathbf{y}_t^*\}$  is bounded. By the optimality of  $(\mathbf{x}_{t+1}^*, \mathbf{p}_{t+1}^*)$  we have

$$\begin{aligned} 0 &\in \partial_{\mathbf{x}} \mathcal{L}(\mathbf{x}_{t+1}^*, \mathbf{p}_{t+1}^*, \mathbf{y}_t^*, \mu_t) = \partial \|\mathbf{x}_{t+1}^*\|_1 - \Phi' \mathbf{y}_{t+1}^*, \\ 0 &= \partial_{\bar{\Gamma}(\mathbf{p})} \mathcal{L}(\mathbf{x}_{t+1}^*, \mathbf{p}_{t+1}^*, \mathbf{y}_t^*, \mu_t) = \bar{\Gamma}(\mathbf{y}_{t+1}^*), \end{aligned}$$

where  $\partial_{\mathbf{x}}$  denotes the partial differential operator with respect to  $\mathbf{x}$  resulting in a set of subgradients. Hence,  $\Phi' \mathbf{y}_{t+1}^* \in \partial \|\mathbf{x}_{t+1}^*\|_1$ . It follows that  $\|\Phi' \mathbf{y}_{t+1}^*\|_{\infty} \leq 1$  and  $\{\mathbf{y}_t^*\}$  is bounded. By  $\mathbf{x}^\dagger \geq \mathcal{L}(\mathbf{x}_{t+1}^*, \mathbf{p}_{t+1}^*, \mathbf{y}_t^*, \mu_t)$ ,

$$\begin{aligned} \|\mathbf{x}_{t+1}^*\|_1 &= \mathcal{L}(\mathbf{x}_{t+1}^*, \mathbf{p}_{t+1}^*, \mathbf{y}_t^*, \mu_t) - \frac{1}{2\mu_t} (\|\mathbf{y}_{t+1}^*\|_2^2 - \|\mathbf{y}_t^*\|_2^2) \\ &\leq \mathbf{x}^\dagger - \frac{1}{2\mu_t} (\|\mathbf{y}_{t+1}^*\|_2^2 - \|\mathbf{y}_t^*\|_2^2). \end{aligned}$$

By  $\{\mathbf{y}_t^*\}$  is bounded,

$$\|\mathbf{x}_{t+1}^*\|_1 \leq \mathbf{x}^\dagger + O(\mu_t^{-1}). \quad (13)$$

For any accumulation point  $\mathbf{x}^*$  of  $\mathbf{x}_t^*$ , without loss of generality, we have  $\mathbf{x}_t^* \rightarrow \mathbf{x}^*$  as  $t \rightarrow +\infty$ . Hence,  $\|\mathbf{x}^*\|_1 \leq \mathbf{x}^\dagger$ . In the mean time,  $\mathbf{p}_{t+1}^* = \Phi \mathbf{x}_{t+1}^* + \mu_t^{-1} (\mathbf{y}_{t+1}^* - \mathbf{y}_t^*) \rightarrow \mathbf{p}^*$  and  $\Phi \mathbf{x}^* = \mathbf{p}^*$  result in that  $(\mathbf{x}^*, \mathbf{p}^*)$  is an optimal solution to  $(\text{BP}^o)$ .

Moreover, by  $\mathbf{x}_{t+1}^* = \Phi' [\mathbf{p}_{t+1}^* - \mu_t^{-1} (\mathbf{y}_{t+1}^* - \mathbf{y}_t^*)]$  and

$$\mathbf{x}^\dagger = \min_{\Phi \mathbf{x} = \mathbf{p}, \Gamma(\mathbf{p}) = \mathbf{b}} \|\mathbf{x}\|_1 = \min_{\mathbf{p}, \Gamma(\mathbf{p}) = \mathbf{b}} \|\Phi' \mathbf{p}\|_1 \leq \|\Phi' \mathbf{p}_{t+1}^*\|_1,$$

we have  $\|\mathbf{x}_{t+1}^*\|_1 \geq \mathbf{x}^\dagger - O(\mu_t^{-1})$ , which establishes the theorem with (13). ■

Algorithm 1 contains, respectively, an inner and an outer iteration loops. Theorem 1 presents only the convergence rate of the outer loop. A natural way to speed up Algorithm 1 is to terminate the inner loop without convergence and use the obtained inner-loop solution as the initialization for the next iteration. This is similar to a continuation strategy and can be realized with reasonably set precision and step size  $\mu_t$  [19]. When the continuation parameter  $\mu_t$  increases very slowly, in a few iterations, the inner loop can produce a solution with high accuracy. In particular, for the purpose of fast and simple computing, we may update the variables in the inner loop only once before stepping into the outer loop operation. This results in a relaxed version of exact ONE-L1 algorithm (eONE-L1), namely relaxed ONE-L1 algorithm (rONE-L1) outlined in Algorithm 2.

Algorithm 2: Relaxed ONE-L1 Algorithm

Input: Expanded orthonormal matrix  $\Phi$  and observed sample data  $\mathbf{b}$ .

1.  $\mathbf{x}_0 = \mathbf{0}; \mathbf{p}_0 = \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix}; \mathbf{y}_0 = \mathbf{0}; \mu_0 > 0; t = 0.$
  2. while not converged do
  3.  $\mathbf{x}_{t+1} = S_{\mu_t}^{-1} (\Phi' (\mathbf{p}_t + \mu_t^{-1} \mathbf{y}_t));$
  4.  $\mathbf{p}_{t+1} = \begin{bmatrix} \mathbf{b} \\ \bar{\Gamma}(\Phi \mathbf{x}_{t+1} - \mu_t^{-1} \mathbf{y}_t) \end{bmatrix};$
  5.  $\mathbf{y}_{t+1} = \mathbf{y}_t + \mu_t (\mathbf{p}_{t+1} - \Phi \mathbf{x}_{t+1});$
  6. choose  $\mu_{t+1} > \mu_t;$
  7. set  $t = t + 1;$
  8. end while
- Output:  $(\mathbf{x}_t, \mathbf{p}_t).$

**Theorem 2:** The iterative solution  $(\mathbf{x}_t, \mathbf{p}_t)$  of Algorithm 2 converges to a feasible solution  $(\mathbf{x}^f, \mathbf{p}^f)$  of  $(\text{BP}^o)$  if  $\sum_{t=1}^{+\infty} \mu_t^{-1} < +\infty$ . It converges at least exponentially to  $(\mathbf{x}^f, \mathbf{p}^f)$  if  $\{\mu_t\}$  is an exponentially increasing sequence.

*Proof:* We show first that sequences  $\{\hat{\mathbf{y}}_t\}$  and  $\{\mathbf{y}_t\}$  are bounded, where  $\hat{\mathbf{y}}_t = \mathbf{y}_{t-1} + \mu_{t-1} (\mathbf{p}_{t-1} - \Phi \mathbf{x}_t)$ . By the optimality of  $\mathbf{x}_{t+1}$  and  $\mathbf{p}_{t+1}$  we have

$$\begin{aligned} 0 &\in \partial_{\mathbf{x}} \mathcal{L}(\mathbf{x}_{t+1}, \mathbf{p}_t, \mathbf{y}_t, \mu_t) = \partial \|\mathbf{x}_{t+1}\|_1 - \Phi' \hat{\mathbf{y}}_{t+1}, \\ 0 &= \partial_{\bar{\Gamma}(\mathbf{p})} \mathcal{L}(\mathbf{x}_{t+1}, \mathbf{p}_{t+1}, \mathbf{y}_t, \mu_t) = \bar{\Gamma}(\mathbf{y}_{t+1}). \end{aligned}$$

Hence,  $\|\Phi' \hat{\mathbf{y}}_{t+1}\|_{\infty} \leq 1$  and it follows that  $\{\hat{\mathbf{y}}_t\}$  is bounded. Since  $\mathbf{y}_{t+1} = \hat{\mathbf{y}}_{t+1} + \mu_t (\mathbf{p}_{t+1} - \mathbf{p}_t)$ , we obtain  $\Gamma(\mathbf{y}_{t+1}) = \Gamma(\hat{\mathbf{y}}_{t+1})$ . This together with  $\bar{\Gamma}(\mathbf{y}_{t+1}) = 0$  results in  $\|\mathbf{y}_{t+1}\|_2 \leq \|\hat{\mathbf{y}}_{t+1}\|_2$  and the boundedness of  $\{\mathbf{y}_t\}$ . By  $\mathbf{p}_{t+1} - \mathbf{p}_t = \mu_t^{-1} (\mathbf{y}_{t+1} - \hat{\mathbf{y}}_{t+1})$ , we have  $\|\mathbf{p}_{t+1} - \mathbf{p}_t\|_2 \leq C \mu_t^{-1}$  with  $C$  being a constant. Then  $\{\mathbf{p}_t\}$  is a Cauchy sequence if  $\sum_{t=1}^{+\infty} \mu_t^{-1} < +\infty$ , resulting in  $\mathbf{p}_t \rightarrow \mathbf{p}^f$  as  $t \rightarrow +\infty$ . In the mean time,  $\mathbf{x}_t \rightarrow \mathbf{x}^f$ ,  $\Phi \mathbf{x}^f = \mathbf{p}^f$ . Hence,  $(\mathbf{x}^f, \mathbf{p}^f)$  is a feasible solution of  $(\text{BP}^o)$ . Suppose that  $\{\mu_t\}$  is an exponentially increasing sequence, i.e.,  $\mu_{t+1} = r \mu_t$  with  $r > 1$ . By the boundedness of  $\{\mathbf{y}_t\}$  and  $\{\hat{\mathbf{y}}_t\}$  we have

$$\begin{aligned} \|\mathbf{p}_t - \mathbf{p}^f\|_2 &= \left\| \sum_{i=t}^{+\infty} (\mathbf{p}_i - \mathbf{p}_{i+1}) \right\|_2 \leq \sum_{i=t}^{+\infty} \|\mathbf{p}_i - \mathbf{p}_{i+1}\|_2 \\ &\leq C \mu_t^{-1} \sum_{i=0}^{+\infty} r^{-i} = O(\mu_t^{-1}). \end{aligned}$$

Hence,  $\{\mathbf{p}_t\}$  converges at least exponentially to  $\mathbf{p}^f$  since  $\{\mu_t^{-1}\}$  exponentially converges to 0, and the same result holds for  $\{\mathbf{x}_t\}$ . ■

**Remark 2:** It is shown in Theorem 2 that faster growth of  $\{\mu_t\}$  can result in faster convergence of  $\{\mathbf{x}_t\}$ . Intuitively, the reduced number of iterations for the inner loop problem  $(SP)$  may result in some error from the optimal solution  $\mathbf{x}_t^*$  of the inner loop. This will likely affect the accuracy of the final solution  $\mathbf{x}^f$  for  $(\text{BP})$ . Therefore, the growth speed of  $\{\mu_t\}$  provides a tradeoff between the convergence speed of the algorithm and the precision of the final solution, which will be illustrated in Section III through numerical simulations.

#### D. Relationship Between rONE-L1 and IST

The studies and applications of IST type algorithms have been very active in recent years because of their concise presentations. This subsection considers the relationship between rONE-L1 and IST. Note that  $\bar{\Gamma}(\mathbf{y}_t) = 0$  in Algorithm 2 and  $\Phi' \Phi = \mathbf{A}' \mathbf{A} + \mathbf{B}' \mathbf{B} = \mathbf{I}$ . After some derivations, it can be shown that the rONE-L1 algorithm is equivalent to the following iteration (starting from  $\mathbf{x}_t = \mathbf{0}$ , as  $t \leq 0$ , and  $\mathbf{z}_t = \mathbf{0}$ , as  $t < 0$ ):

$$\begin{aligned} \mathbf{x}_{t+1} &= S_{\lambda_t} (\mathbf{x}_t + \mathbf{A}' \mathbf{z}_t), \\ \mathbf{z}_t &= \mathbf{b} - \mathbf{A} [(1 + \kappa_t) \mathbf{x}_t - \kappa_t \mathbf{x}_{t-1}] + \kappa_t \mathbf{z}_{t-1}, \end{aligned} \quad (14)$$

where  $\lambda_t = \mu_t^{-1}$  and  $\kappa_t = \frac{\mu_{t-1}}{\mu_t}$ . Compared with the general form of IST in (2), one more term  $\kappa_t \mathbf{z}_{t-1}$  is added when computing the current residual  $\mathbf{z}_t$  in rONE-L1. Moreover, a weighted sum  $(1 + \kappa_t) \mathbf{x}_t - \kappa_t \mathbf{x}_{t-1}$  is used instead of the current solution  $\mathbf{x}_t$ . It will be shown later that these two changes essentially improve the sparsity-undersampling tradeoff.

**Remark 3:** Equations in (14) show that the expansion from the partially orthonormal matrix  $\mathbf{A}$  to orthonormal  $\Phi$  is not at

all involved in the actual implementation and computation of rONE-L1. The same claim also holds for eONE-L1 algorithm. Nevertheless, the orthonormal expansion is a key instrumentation in the derivation and analysis of Algorithms 1 and 2.

### E. Implementation Details

As noted in Remark 3, the expansion from  $\mathbf{A}$  to  $\Phi$  is not involved in the computing of ONE-L1 algorithms. In our implementations, we consider using exponentially increasing  $\mu_t$ , i.e., fixing  $r > 1$  and  $\mu_t = r^t \mu_0$ . Let  $Q_\alpha(\cdot)$  be an  $\alpha$ -quantile operator and  $\mu_0 = 1/Q_\alpha(|\mathbf{A}'\mathbf{b}|)$ , with  $|\cdot|$  applying to the vector variable elementwise,  $\mu_0^{-1}$  being the threshold in the first iteration and  $\alpha = 0.99$ . In eONE-L1, a large  $r$  can speed up the convergence of the outer loop iteration according to Theorem 1. However, simulations show that a larger  $r$  can result in more iterations in the inner loop. We use  $r = 1 + n/N$  as default. In rONE-L1, the value of  $r$  provides a tradeoff between the convergence speed of the algorithm and the precision of the final solution. Our recommendation of  $r$  to achieve the optimal sparsity-undersampling tradeoff is  $r = \min(1 + 0.04n/N, 1.02)$ , which will be illustrated in Section III-A.

An iterative algorithm needs a termination criterion. The eONE-L1 algorithm is considered converged if  $\frac{\|\mathbf{A}\mathbf{x}_t^* - \mathbf{b}\|_2}{\|\mathbf{b}\|_2} < \tau_1$  with  $\tau_1$  being a user-defined tolerance. The inner iteration is considered converged if  $\frac{\|\mathbf{x}_t^{j+1} - \mathbf{x}_t^j\|_2}{\|\mathbf{x}_t^j\|_2} < \tau_2$ . In our implementation, the default values are  $(\tau_1, \tau_2) = (10^{-5}, 10^{-6})$ . The rONE-L1 algorithm is considered converged if  $\frac{\|\mathbf{A}\mathbf{x}_t - \mathbf{b}\|_2}{\|\mathbf{b}\|_2} < \tau$ , with  $\tau = 10^{-5}$  as default.

## III. NUMERICAL SIMULATIONS

### A. Sparsity-Undersampling Tradeoff

This subsection considers the sparsity-undersampling tradeoff of rONE-L1 in the case of strictly sparse signals and noise-free measurements. Phase transition is a measure of the sparsity-undersampling tradeoff in this case. Let the sampling ratio be  $\delta = n/N$  and the sparsity ratio be  $\rho = k/n$ , where  $k$  is a measure of sparsity of  $\mathbf{x}$ , and we call that  $\mathbf{x}$  is  $k$ -sparse if at most  $k$  entries of  $\mathbf{x}$  are nonzero. As  $k, n, N \rightarrow \infty$  with fixed  $\delta$  and  $\rho$ , the behavior of the phase transition of (BP) is controlled by  $(\delta, \rho)$  [20], [21]. We denote this theoretical curve by  $\rho = \rho_T(\delta)$ , which is plotted in Fig 1.

We estimate the phase transition of rONE-L1 using a Monte Carlo method as in [17], [22]. Two matrix ensembles are considered, including Gaussian with  $N = 1000$  and partial-DCT with  $N = 1024$ . Here the finite- $N$  phase transition is defined as the value of  $\rho$  at which the success probability to recover the original signal is 50%. We consider 33 equispaced values of  $\delta$  in  $\{0.02, 0.05, \dots, 0.98\}$ . For each  $\delta$ , 21 equispaced values of  $\rho$  are generated in the interval  $[\rho_T(\delta) - 0.1, \rho_T(\delta) + 0.1]$ . Then  $M = 20$  random problem instances are generated and solved with respect to each combination of  $(\delta, \rho)$ , where  $n = \lceil \delta N \rceil$ ,  $k = \lceil \rho n \rceil$ , and nonzero entries of sparse signals are generated from the standard Gaussian distribution. Success is declared if the relative root mean squared error (relative

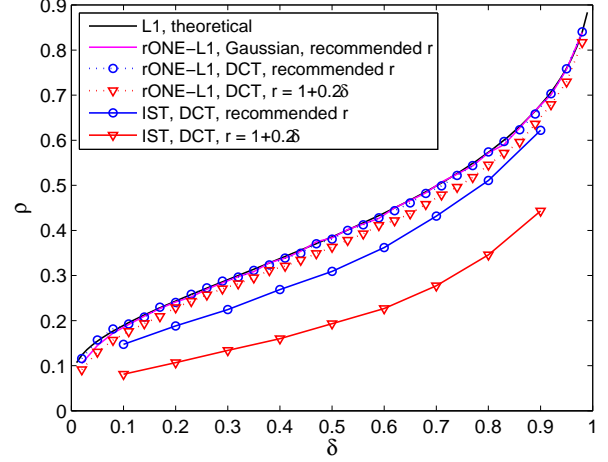


Fig. 1. Observed phase transitions of rONE-L1, and comparison with those of IST. Note that, 1) the observed phase transitions of rONE-L1 with the recommended  $r$  strongly agree with the theoretical calculation based on (BP); 2) rONE-L1 has significantly enlarged success phases compared with IST.

RMSE)  $\frac{\|\hat{\mathbf{x}} - \mathbf{x}^o\|_2}{\|\mathbf{x}^o\|_2} < 10^{-4}$ , where  $\hat{\mathbf{x}}$  is the recovered signal. The number of success among  $M$  experiments is recorded. Finally, a generalized linear model is used to estimate the phase transition as in [22].

The experiment result is presented in Fig. 1. The observed phase transitions using the recommended value of  $r$  strongly agree with the theoretical result of (BP). It shows that the rONE-L1 algorithm has the optimal sparsity-undersampling tradeoff in the sense of  $\ell_1$  minimization.

### B. Comparison with IST

The rONE-L1 algorithm can be considered as a modified version of IST in (2). In this subsection we compare the sparsity-undersampling tradeoff and speed of these two algorithms. A similar method is adopted to estimate the phase transition of IST, which is implemented using the same parameter values as rONE-L1. Only nine values of  $\delta$  in  $\{0.1, 0.2, \dots, 0.9\}$  are considered with the partial-DCT matrix ensemble for time consideration. Another choice of  $r = 1 + 0.2\delta$  is considered besides the recommended one. Correspondingly, the phase transition of rONE-L1 with  $r = 1 + 0.2\delta$  is also estimated.

The observed phase transitions are shown in Fig. 1. As a modified version of IST, obviously, rONE-L1 makes a great improvement over IST in the sparsity-undersampling tradeoff. Meanwhile, comparison of the averaged number of iterations of the two algorithms shows that rONE-L1 is also faster than IST. For example, as  $\delta = 0.2$  and the recommended  $r$  is used, rONE-L1 is about 6 times faster than IST.

### C. Comparison with AMP, FPC-AS and NESTA in Noise-free Case

In this subsection, we report numerical simulation results comparing rONE-L1 with state-of-the-art algorithms, including AMP, FPC-AS and NESTA, in the case of sparse signals

TABLE I  
COMPARISON RESULTS OF ONE-L1 ALGORITHMS WITH  
STATE-OF-THE-ART METHODS

$\rho$	Method	# calls $\mathbf{A}$ & $\mathbf{A}'$	CPU time (s)	Error ( $10^{-5}$ )
0.1	eONE-L1	1819 (1522,2054)*	5.62 (4.67,6.52)	0.42 (0.11,0.94)
	rONE-L1	515.4 (286,954)	2.14 (1.19,3.92)	1.08 (0.53,1.30)
	AMP	222.7 (216,234)	0.80 (0.76,0.86)	1.02 (0.85,1.15)
	FPC-AS	150.2 (135,170)	0.50 (0.44,0.56)	1.13 (1.07,1.23)
	NESTA	468.9 (458,484)	2.70 (2.55,2.98)	1.05 (0.99,1.13)
0.22	eONE-L1	9038 (7270,11194)	28.5 (22.0,35.8)	1.87 (0.46,2.66)
	rONE-L1	722.3 (440,972)	2.61 (1.63,3.93)	1.80 (1.37,3.05)
	AMP	1708 (1150,2252)	6.21 (4.19,9.11)	10.5 (6.96,15.8)
	FPC-AS	589.4 (476,803)	2.10 (1.65,2.80)	1.96 (1.46,3.60)
	NESTA	1084 (890,1244)	6.47 (5.22,7.50)	2.90 (1.62,3.98)

\*Three numbers are averaged, minimum and maximum values, respectively.

and noise-free measurements. Our experiments used FPC-AS v.1.21, NESTA v.1.1, and AMP codes provided by the author. We choose parameter values for FPC-AS and NESTA such that each method produces a solution with approximately the same precision as that produced by rONE-L1. In our experiments we consider the recovery of exactly sparse signals from partial-DCT measurements. We set  $N = 2^{14}$  and  $\delta = 0.2$ , and an ‘easy’ case where  $\rho = 0.1$  and a ‘hard’ case where  $\rho = 0.22$  are considered, respectively.<sup>1</sup> Twenty random problems are created and solved for each algorithm with each combination of  $(\delta, \rho)$ , and the minimum, maximum and averaged relative RMSE, number of calls of  $\mathbf{A}$  and  $\mathbf{A}'$ , and CPU time usages are recorded. All experiments are carried on Matlab v.7.7.0 on a PC with a Windows XP system and a 3GHz CPU. Default parameter values are used in eONE-L1 and rONE-L1.

**AMP:** terminating if  $\frac{\|\mathbf{A}\mathbf{x}_t - \mathbf{b}\|_2}{\|\mathbf{b}\|_2} < 10^{-5}$ .

**FPC-AS:**  $\lambda = 2 \times 10^{-6}$  and  $gtol = 1 \times 10^{-6}$ , where  $gtol$  is the termination criterion on the maximum norm of sub-gradient. FPC-AS solves the problem  $(\mathbf{QP}_\lambda)$ .

**NESTA:**  $\lambda = 2 \times 10^{-6}$ ,  $\epsilon = 0$  and the termination criterion  $tolvar = 1 \times 10^{-8}$ . NESTA solves  $(\mathbf{BP}_\epsilon)$  using the Nesterov algorithm [11], with continuation.

Our experiment results are presented in Table I. In both ‘easy’ and ‘hard’ cases, rONE-L1 is much faster than eONE-L1. In the ‘easy’ case, the proposed rONE-L1 algorithm takes the most number of calls of  $\mathbf{A}$  and  $\mathbf{A}'$ , except that of eONE-L1, due to a conservative setting of  $r$ . But this number of calls (515.4) is very close to that of NESTA (468.9), and furthermore, the CPU time usage of rONE-L1 (2.14 s) is less than that of NESTA (2.70 s) because of its concise implementation. In the ‘hard’ case, rONE-L1 has the second best performance with significantly less CPU time than that of AMP and NESTA. AMP has the second worst CPU time and the worst accuracy as the dynamic threshold in each iteration depends on the mean squared error of the current iterative solution, which cannot be calculated exactly in the implementation.

<sup>1</sup>Here ‘easy’ and ‘hard’ refer to the difficulty degree in recovering a sparse signal from a specific number of measurements. The setting  $(\delta, \rho) = (0.2, 0.22)$  is close to the phase transition of  $(\mathbf{BP})$ .

#### D. A Practical Example

This subsection demonstrates the efficiency of rONE-L1 in the general CS where the signal of interest is approximately sparse and measurements are contaminated with noise. We seek to reconstruct the Mondrian image of size  $256 \times 256$ , shown in Fig. 2, from its noise-contaminated partial-DCT coefficients. This image presents a challenge as its wavelet expansion is compressible but not exactly sparse. The sampling pattern, which is inspired by magnetic resonance imaging (MRI) and is shown in Fig. 2, is adopted as most energy of the image concentrates at low-frequency components after the DCT transform. The measurement vector  $\mathbf{b}$  contains  $n = 7419$  DCT measurements ( $\delta = 0.113$ ). White Gaussian noise with standard deviation  $\sigma = 1$  is then added. We set  $\epsilon = \sqrt{n + 2\sqrt{2}n\sigma}$ . Haar wavelet with a decomposition level 4 is chosen as the sparsifying transform  $\mathcal{W}$ . Hence, the problem to be solved is  $(\mathbf{BP}_\epsilon)$  with  $\mathbf{A} = \mathcal{F}_p \mathcal{W}'$ , where  $\mathcal{F}_p$  is the partial-DCT transform. The reconstructed image is  $\hat{\mathbf{H}} = \mathcal{W}'\hat{\mathbf{x}}$  with  $\hat{\mathbf{x}}$  being the reconstructed wavelet coefficients and reconstruction error is calculated as  $\frac{\|\hat{\mathbf{H}} - \mathbf{H}^o\|_F}{\|\mathbf{H}^o\|_F}$ , where  $\mathbf{H}^o$  is the original image and  $\|\cdot\|_F$  denotes the Frobenius norm. We compare the performance of rONE-L1 with NESTA and FPC-AS.

*Remark 4:* AMP is omitted for its poor performance in this approximately-sparse-signal case. For AMP, the value of the dynamic threshold  $\lambda_t$  and the term  $\|x_t\|_0$  in (3) depend on the condition that the signal to reconstruct is strictly sparse.

In such a noisy measurement case, an exact solution for  $(\mathbf{BP}_\epsilon)$  is not sought after in the rONE-L1 simulation. The computation of the rONE-L1 algorithm is set to terminate if  $\frac{\|\mathbf{A}\mathbf{x}_t - \mathbf{b}\|_2}{\|\mathbf{b}\|_2} \leq \tau = \frac{\epsilon}{\|\mathbf{b}\|_2}$ , i.e., rONE-L1 outputs the first  $\mathbf{x}_t$  when it becomes a feasible solution of  $(\mathbf{BP}_\epsilon)$ .

**FPC-AS:**  $\lambda = 1 \times 10^{-3}$ ,  $gtol = 1 \times 10^{-3}$ ,  $gtol\_scale\_x = 1 \times 10^{-6}$  and the maximum number of iterations for subspace optimization  $sub\_maxitr = 10$ . The parameters are set according to [15, Section 4.4].

**NESTA:**  $\lambda = 1 \times 10^{-4}$ , and  $tolvar = 1 \times 10^{-6}$ . The parameters are tuned to achieve the minimum reconstruction error.

Fig. 2 shows the experiment results where rONE-L1, FPC-AS and NESTA produce faithful reconstructions of the original image. The rONE-L1 algorithm produces a reconstruction error (0.0741) lower than that of FPC-AS (0.0809) with comparable computation times (11.1 s and 11.4 s, respectively). While NESTA results in a slightly lower reconstruction error (0.0649), it incurs about twice more computation time (29.4 s).

#### IV. CONCLUSION

In this work, we have presented novel algorithms to solve the basis pursuit problem for noiseless CS. The proposed rONE-L1 algorithm, based on the augmented Lagrange multiplier method and heuristic simplification, can be considered as a modified IST with an aggressive continuation strategy. The following two cases of CS have been studied: 1) exact reconstruction of sparse signals from noise-free measurements, and 2) reconstruction of approximately sparse signals from noise-contaminated measurements. The proposed rONE-L1

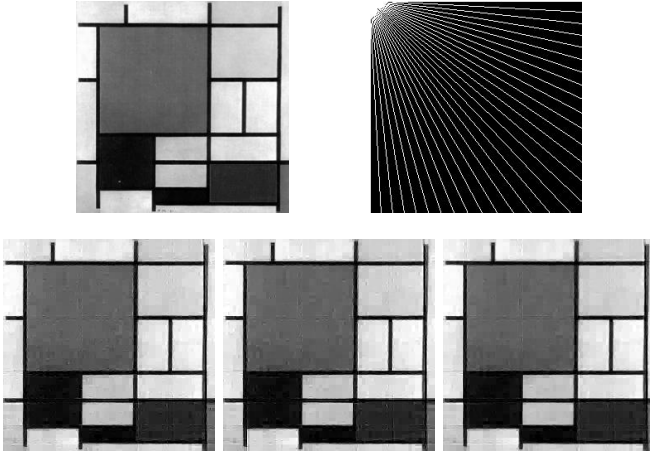


Fig. 2. An example of 2D image reconstruction from noise-contaminated partial-DCT measurements. Upper left: original Mondrian image; upper right: sampling pattern. The lower three are reconstructed images respectively by rONE-L1 (lower left, error: 0.0741, time: 11.1 s), FPC-AS (lower middle, error: 0.0809, time: 11.4 s) and NESTA (lower right, error: 0.0649, time: 29.4 s).

outperforms AMP, which is a well-known IST type algorithm, in Case 2 and also in Case 1 when the setting of  $(\delta, \rho)$  is close to the phase transition of basis pursuit. It is faster than NESTA in both Case 1 and Case 2. The numerical experiments further show that rONE-L1 outperforms FPC-AS in Case 2. Apart from the high computational efficiency and accurate reconstruction result, another useful property of rONE-L1 is its ease of parameter tuning. It only needs to set a termination criterion  $\tau$  if the recommended  $r$  is used and the value of  $\tau$  is explicit in Case 2. While this correspondence focuses on reconstruction of real-valued signals, it is straightforward to apply ONE-L1 algorithms to the reconstruction of complex-valued signals [23]. More rigorous analysis of rONE-L1 is currently under investigation.

#### ACKNOWLEDGMENT

The authors are grateful to the anonymous reviewers for helpful comments. Z. Yang wishes to thank Arian Maleki for providing the AMP codes. The Matlab codes of ONE-L1 algorithms are available at <http://sites.google.com/site/zaiyang0248>.

#### REFERENCES

- [1] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Info. Theo.*, vol. 52, no. 2, pp. 489–509, 2006.
- [2] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, pp. 969–985, 2007.
- [3] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [4] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.
- [5] K. Bredies and D. Lorenz, "Linear convergence of iterative soft-thresholding," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 813–837, 2008.
- [6] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An Interior-Point Method for Large-Scale  $\ell_1$ -Regularized Least Squares," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 1, no. 4, pp. 606–617, 2008.

- [7] M. Lustig, D. Donoho, and J. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.
- [8] E. Hale, W. Yin, and Y. Zhang, "A fixed-point continuation method for  $\ell_1$ -regularized minimization with applications to compressed sensing," *CAAM TR07-07, Rice University*, 2007.
- [9] E. Candès and J. Romberg, " $\ell_1$ -magic: Recovery of sparse signals via convex programming," URL: [www.acm.caltech.edu/l1magic/downloads/l1magic.pdf](http://www.acm.caltech.edu/l1magic/downloads/l1magic.pdf).
- [10] S. Becker, J. Bobin, and E. Candès, "NESTA: A fast and accurate first-order method for sparse recovery," *SIAM J. Imaging Sciences*, vol. 4, no. 1, pp. 1–39, 2011.
- [11] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematical Programming*, vol. 103, no. 1, pp. 127–152, 2005.
- [12] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.
- [13] D. Donoho, Y. Tsaig, I. Drori, and J. Starck, "Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit," *submitted to IEEE Trans. on Signal Processing*. Available at: [www.cs.tau.ac.il/~idrori/StOMP.pdf](http://www.cs.tau.ac.il/~idrori/StOMP.pdf), 2006.
- [14] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.
- [15] Z. Wen, W. Yin, D. Goldfarb, and Y. Zhang, "A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization and continuation," *SIAM Journal on Scientific Computing*, vol. 32, no. 4, pp. 1832–1857, 2010.
- [16] A. Maleki and D. Donoho, "Optimally tuned iterative thresholding algorithms for compressed sensing," *IEEE J. Select. Areas Signal Processing*, vol. 4, pp. 330–341, 2010.
- [17] D. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18 914–18 919, 2009.
- [18] J. Nocedal and S. Wright, *Numerical Optimization*. New York: Springer verlag, 2006.
- [19] D. Bertsekas, *Constrained optimization and Lagrange multiplier methods*. Boston: Academic Press, 1982.
- [20] D. Donoho and J. Tanner, "Counting the faces of randomly-projected hypercubes and orthants, with applications," *Discrete and Computational Geometry*, vol. 43, no. 3, pp. 522–541, 2010.
- [21] M. Stojnic, "Various thresholds for  $\ell_1$ -optimization in compressed sensing," *arXiv:0907.3666*, 2009.
- [22] D. Donoho and J. Tanner, "Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing," *Philosophical Transactions of the Royal Society A*, vol. 367, no. 1906, pp. 4273–4293, 2009.
- [23] Z. Yang and C. Zhang, "Sparsity-undersampling tradeoff of compressed sensing in the complex domain," in *Proceedings of ICASSP 2011*, 2011.